

**Amendments to the Claims:**

A listing of the entire set of pending claims (including amendments to the claims, if any) is submitted herewith per 37 CFR 1.121. This listing of claims will replace all prior versions, and listings, of claims in the application.

**Listing of Claims:**

1. (Original) A method of generating a common secret between a first party and a second party, in which the first party holds a value  $p_1$  and a symmetrical polynomial  $P(x, y)$  fixed in the first argument by the value  $p_1$ , and the first party performs the steps of sending the value  $p_1$  to the second party, receiving a value  $p_2$  from the second party and calculating the common secret  $S_1$  by evaluating the polynomial  $P(p_1, y)$  in  $p_2$ , characterized in that the first party additionally holds a value  $q_1$  and a symmetrical polynomial  $Q(x, z)$  fixed in the first argument by the value  $q_1$ , and further performs the steps of sending  $q_1$  to the second party, receiving a value  $q_2$  from the second party and calculating the secret  $S_1$  as  $S_1 = Q(q_1, q_2) \cdot P(p_1, p_2)$ .

2. (Original) The method of claim 1, in which the first party further performs the steps of obtaining a random number  $r_1$ , calculating  $r_1 \cdot q_1$ , sending  $r_1 \cdot q_1$  to the second party, receiving  $r_2 \cdot q_2$  from the second party and calculating the secret  $S_1$  as  $S_1 = Q(q_1, r_1 \cdot r_2 \cdot q_2) \cdot P(p_1, p_2)$ .

3. (Original) The method of claim 2, in which the first party holds the value  $q_1$  multiplied by an arbitrarily chosen value  $r$ , and the product  $Q(q_1, z)P(p_1, y)$  instead of the individual polynomials  $P(p_1, y)$  and  $Q(q_1, z)$ , and the first party performs the steps of calculating  $r_1 \cdot r \cdot q_1$ , sending  $r_1 \cdot r \cdot q_1$  to the second party, receiving  $r_2 \cdot r \cdot q_2$  from the second party and calculating the secret  $S_1$  as  $S_1 = Q(q_1, r_1 \cdot r_2 \cdot r \cdot q_2) \cdot P(p_1, p_2)$ .

4. (Original) The method of claim 1, in which the second party holds a value  $p_2$  and a value  $q_2$ , the symmetrical polynomial  $P(x, y)$  fixed in the first argument by the value  $p_2$ , the symmetrical polynomial  $Q(x, z)$  fixed in the first argument by the value  $q_2$ , and the second party performs the steps of sending  $q_2$  to the first party, receiving  $q_1$  from the first party and calculating a secret  $S_2$  as  $S_2 = Q(q_2, q_1) \cdot P(p_2, p_1)$ , whereby the common secret has been generated if the secret  $S_2$  equals the secret  $S_1$ .

5. (Original) The method of claim 1, in which a trusted third party performs the steps of choosing a symmetric  $(n+1) \times (n+1)$  matrix  $T$ , constructing the polynomial  $P$  using entries from the matrix  $T$  as respective coefficients of the polynomial  $P$ , constructing the polynomial  $Q(x, y)$ , choosing the value  $p_1$ , the value  $p_2$ , the value  $q_1$  and the value  $q_2$ , sending the value  $p_1$ , the value  $q_1$ , the polynomial  $P(x, y)$  fixed in the first argument by the value  $p_1$  and the polynomial  $Q(x, z)$  fixed in the first argument by the value  $q_1$  to the first party, and sending the value  $p_2$ , the value  $q_2$ , the polynomial  $P(x, y)$  fixed in the first argument by the value  $p_2$  and the polynomial  $Q(x, z)$  fixed in the first argument by the value  $q_2$  to the second party

6. (Original) The method of claim 5, in which the trusted third party further arbitrarily chooses a value  $r$ , sends the value  $r \cdot q_1$  instead of the value  $q_1$  and the product  $Q(q_1, z)P(p_1, y)$  instead of the individual polynomials  $P(p_1, y)$  and  $Q(q_1, z)$  to the first party and sends the value  $r \cdot q_2$  instead of the value  $q_2$  and the product  $Q(q_2, z)P(p_2, y)$  instead of the individual polynomials  $P(p_2, y)$  and  $Q(q_2, z)$  to the second party.

7. (Original) The method of claim 5, in which the trusted third party further performs the steps of

choosing a set comprising  $m$  values  $p_i$ , including the values  $p_1$  and  $p_2$ ,

calculating a space  $\mathbf{A}$  from the tensor products  $\bar{p}_i^{V'} \otimes \bar{p}_j^{V'}$  of the Vandermonde

vectors  $\bar{p}_i^{V'}$  built from the set of values  $p_i$ ,

choosing a vector  $\bar{\gamma}_1$  and a vector  $\bar{\gamma}_2$  from the perpendicular space  $\mathbf{A}^\perp$  of the space  $\mathbf{A}$ , constructing a matrix  $T_{\Gamma_1} = T + \Gamma_1$  from the vector  $\bar{\gamma}_1$  and a matrix  $T_{\Gamma_2} = T + \Gamma_2$  from the vector  $\bar{\gamma}_2$ , constructing a polynomial  $P^{\Gamma_1}(x, y)$  using entries from the matrix  $T_{\Gamma_1}$  and sending the polynomial  $P^{\Gamma_1}(x, y)$  fixed in the first argument by the value  $p_1$  to the first party, and

constructing a polynomial  $P^{\Gamma_2}(x, y)$  using entries from the matrix  $T_{\Gamma_2}$  and sending the polynomial  $P^{\Gamma_2}(x, y)$  fixed in the first argument by the value  $p_2$  to the second party.

8. (Original) The method of claim 5, in which a number  $m'$  of values  $p_i$ , and  $m' < m$ , are distributed to additional parties.

9. (Original) The method of claim 1, in which the first party and the second party use a non-linear function on the generated secret S1 and S2, respectively, before using it as a secret key in further communications.

10. (Original) The method of claim 9 in which a one-way hash function is applied to the generated secrets S1 and S2.

11. (Original) The method of claim 9 in which a non-linear function in the form of a polynomial is applied to the generated secrets S1 and S2.

12. (Original) The method of claim 1, further comprising the step of verifying that the second party knows the secret  $S_1$ .

13. (Original) The method of claim 12, in which the first party subsequently applies a zero-knowledge protocol to verify that the second party knows the secret  $S_1$ .

14. (Original) The method of claim 12, in which the first party subsequently applies a commitment-based protocol to verify that the second party knows the secret  $S_1$ .

15. (Original) The method of claim 14, in which the second party uses a symmetric cipher to encrypt a random challenge, and sends the encrypted random challenge to the first party and the first party subsequently uses the same symmetric cipher as a commit function to commit himself to a decryption of the encrypted random challenge.

16. (Currently amended) A system ~~(100)~~ comprising a first party ~~(P)~~, a second party ~~(V)~~ and a trusted third party ~~(TTP)~~, that is arranged to execute the method of claim 4 generate a common secret between the first party and the second party, in which the first party holds a value  $p_1$  and a symmetrical polynomial  $P(x,y)$  fixed in the first argument by the value  $p_1$ , and the first party performs the steps of sending the value  $p_1$  to the second party, receiving a value  $p_2$  from the second party and calculating the common secret  $S_1$  by evaluating the polynomial  $P(p_1, y)$  in  $p_2$ .

wherein the first party additionally holds a value  $q_1$  and a symmetrical polynomial  $Q(x, z)$  fixed in the first argument by the value  $q_1$ , and further performs the steps of sending  $q_1$  to the second party, receiving a value  $q_2$  from the second party and calculating the secret  $S_1$  as  $S_1=Q(q_1, q_2) \cdot P(p_1, p_2)$ .

17. (Original) A device (P) arranged to operate as the first party and/or as the second party in the system of claim 16.

18. (Original) The device of claim 17, comprising storage means (303) for storing the polynomial P and the polynomial Q in the form of their respective coefficients.

19. (Currently amended) A computer program product for causing one or more processors to ~~execute the method of claim 1~~ generate a common secret between a first party and a second party, in which the first party holds a value  $p_1$  and a symmetrical polynomial  $P(x,y)$  fixed in the first argument by the value  $p_1$ , and the first party performs the steps of sending the value  $p_1$  to the second party, receiving a value  $p_2$  from the second party and calculating the common secret  $S_1$  by evaluating the polynomial  $P(p_1, y)$  in  $p_2$ ,

wherein the first party additionally holds a value  $q_1$  and a symmetrical polynomial  $Q(x, z)$  fixed in the first argument by the value  $q_1$ , and further performs the steps of sending  $q_1$  to the second party, receiving a value  $q_2$  from the second party and calculating the secret  $S_1$  as  $S_1=Q(q_1, q_2) \cdot P(p_1, p_2)$ .

20. (New) The system of claim 16, wherein the second party holds a value  $p_2$  and a value  $q_2$ , the symmetrical polynomial  $P(x, y)$  fixed in the first argument by the value  $p_2$ , the symmetrical polynomial  $Q(x, z)$  fixed in the first argument by the value  $q_2$ , and the second party performs the steps of sending  $q_2$  to the first party, receiving  $q_1$  from the first party and calculating a secret  $S_2$  as  $S_2=Q(q_2, q_1) \cdot P(p_2, p_1)$ , whereby the common secret has been generated if the secret  $S_2$  equals the secret  $S_1$ .